Experimental Evaluation of Subject Matter Expert-oriented Knowledge Base Authoring Tools

Robert Schrag[1], Mike Pool[1], Vinay Chaudhri[2], Robert C. Kahlert[3], Joshua Powers[1], Paul Cohen[4], Julie Fitzgerald[1], and Sunil Mishra[4]

1) Information Extraction & Transport (IET), Inc.
1911 North Fort Myer Drive #600
Arlington, Virginia 22209
last-name@iet.com

2) SRI International
333 Ravenswood Avenue
Menlo Park, CA 94025
{vinay, smishra}@ai.sri.com

3) Cycorp, Inc.
3721 Executive Center Drive #100
Austin, Texas 78731
rck@cyc.com

4) Department of Computer Science
University of Massachusetts
Amherst, MA 01003
cohen@cs.umass.edu

ABSTRACT

We describe a large-scale experiment in which non-artificial intelligence subject matter experts (SMEs)—with neither artificial intelligence background nor extensive training in the task—author knowledge bases (KBs) following a challenge problem specification with a strong question-answering component. As a reference for comparison, professional knowledge engineers (KEs) author KBs following the same specification. This paper concentrates on the design of the experiment and its results—the evaluation of SME- and KE-authored KBs and SME-oriented authoring tools.

Evaluation is in terms of quantitative subjective (functional performance) metrics and objective (knowledge reuse) metrics that we define and apply, as well as in terms of subjective qualitative assessment using several sources. While all evaluation styles are useful individually and exhibit collective power, we find that subjective qualitative evaluation affords us insights of greatest leverage for future system/process design. One practical conclusion is that large-scale KB development may best be supported by "mixed-skills" teams of SMEs and KEs collaborating synergistically, rather than by SMEs forced to work alone.

KEYWORDS: *knowledge acquisition, evaluation*

1 INTRODUCTION

The authors are engaged in a joint research program—Rapid Knowledge Formation (RKF)—to develop and evaluate technology to enable SMEs to build very large KBs. Two teams respond to challenge problems posed by an independent evaluator. We report on a large-scale evaluation conducted during the summer of 2001.

The RKF teams are led by Cycorp and SRI International. The independent evaluator is IET. More comprehensive information about the evaluation—including a full challenge problem specification—is available at http://www.iet.com/Projects/RKF/. For more general program information, see http://reliant.teknowledge.com/RKF/.

In the remainder of this paper, we first outline our approach to evaluating KBs. Then we describe the "textbook knowledge" challenge problem (TKCP) presented to SMEs and KEs for KB authoring, teams' tools, experimental procedures, and results from each style of evaluation. We close with discussion/conclusions.

2 KB EVALUATION APPROACH

We consider KB evaluation along three dimensions: functional performance (with subjective metrics), economics (with objective metrics), and intrinsic quality (subjective and non-metric). Here we elaborate on these dimensions and our methodology. In a later section we describe results.

To evaluate functional performance, we follow Cohen et al. [3] in posing test questions (TQs) to authored KBs and scoring their answers against defined criteria. Our criteria fall into three major categories: Representation (with criteria Query Formulation, Term Quality, and Compositionality), Answer (with criterion Correctness, only), and Explanation (with criteria Content Adequacy, Content Relevance, Intelligibility, and Organization). While the Answer category obviously addresses a KB's functional performance, we argue that high-quality question representations and explanations also confer valuable (input- and output-oriented) functionality to KBs.

To evaluate economics, we follow Cohen et al. [2] in addressing reuse—the extent to which knowledge created earlier is exploited in the creation of subsequent knowledge. We require that authored knowledge (including constants and axioms) bear labels of authorship and creation time. Other things being equal, greater reuse is considered more economical.

Others—[4], [5]—have suggested (without employing, to our knowledge, in large-scale comparative evaluation) qualitative criteria for assessing intrinsic properties of KBs

and ontologies. Inspired by these, we formed a KB Quality Review Panel from among technology providers and evaluators to assess the following properties: Clarity or Style, Maintainability or Reusability, Correctness or Accuracy, Appropriate Generality, Appropriate Organization, and Logical Propriety. While we discussed making this evaluation quantitative (by adapting our Functional Performance scoring methodology described below), the Panel ultimately agreed that free-form commenting along these dimensions would be the most fruitful initial step.

We drew on two other sources, besides the Panel, in our subjective qualitative evaluation: post-evaluation SME survey responses and evaluator observations. Findings regarding RKF tools' strengths and weaknesses were consistent across all three sources. RKF tool developers have taken these results seriously and have begun appropriate modifications to their tools.

2.1 Additional Related Work

The series of Knowledge Acquisition Workshops (KAWs)[1] has emphasized the evaluation of generic problem-solving methods (PSMs) and performance on the associated problem-solving tasks more than that of knowledge for its own sake. This appears to reflect a difference in emphasis or philosophy: whereas the KAW community has focused on the PSM as the primary reusable artifact, the RKF community has focused on KBs themselves as reusable artifacts that should, in principle, be applicable to any problem-solving task.

3 TEXTBOOK KNOWLEDGE CHALLENGE PROBLEM

The TKCP's KB authoring task is to:

1. Capture knowledge about DNA transcription and translation from about ten pages of an introductory undergraduate molecular biology textbook for non-majors [1];
2. Ensure that the authored KBs are capable of correctly answering test questions about the subject material, (extending or revising KBs as necessary).

We chose a textbook source because it serves as a circumscribed reference that offers an intuitively justified basis for required KB content scope. We chose molecular biology because it is a largely descriptive science and because it is of interest to the sponsor. We chose [1] because it largely eschews description of laboratory procedures or scientific history in favor of material phenomena.

The TQs were consistent in difficulty with TQs typically found on Web-available quizzes on molecular biology. Questions appearing in the textbook itself typically required representation of (e.g., hypothetical/counter-factual) situations that were entirely novel compared to the basic material presented in the text. These were judged by the RKF community to be unsuitable (too difficult) for use in evaluation of current SME-oriented KB authoring technology. The TQs were similar in style and difficulty to IET-created sample questions (SQs) covering material in earlier chapters of the textbook. SQs were provided to teams before the evaluation. TQs were not so disclosed.

Besides the primary KB authoring tools described in the following section, RKF teams were required to include facilities for SMEs to pose TQs and to package their answers for evaluation. They also were required to prepare various instrumentation capabilities in support of metrics computations.

Teams' tools included substantial TKCP-relevant knowledge before they were handed off to SMEs. Given the premise that a large, general/reusable KB facilitates the construction of more specific KBs, teams were allowed to "prime the pump" of knowledge development by seeding KBs with prerequisite (e.g., pertaining to earlier—largely review—textbook chapters) and background (including high-level/abstract) knowledge or reasoning abilities deemed appropriate (according to defined ground rules) to support the authoring of the textbook's target knowledge.

4 TOOLS UNDER EVALUATION[2]

Cycorp's "KRAKEN" tools are supported by a substantial KB based on a higher-order formal predicate logic. The key strategies of SME-oriented KB interaction are natural language (NL) presentation and a knowledge-driven acquisition dialog with limited NL understanding. The KB includes thousands of predicates and understands thousands of English verbs. Cycorp's approach might be described as maximalistic, domain-pluralistic, and conceptually precise. The KRAKEN tools aim to exploit (as leverage) a substantial KB to bring SMEs past an otherwise-steep learning curve by productive collaboration in this sophisticated knowledge representation milieu.

SRI's "SHAKEN" tools are supported by a relatively sparse KB based on the frame formalism. The key strategy of SME-oriented interaction is graphical assembly of components. The KB includes a few hundred predicates serving as conceptual primitives (the components). SRI's approach might be described as minimalistic, domain-universal, and conceptually coarse. The SHAKEN tools may be seen as skirting traditional knowledge representation complexity by presenting an entirely new metaphor with great intuitive appeal.

5 EXPERIMENTAL PROCEDURES

IET collaborated with George Mason University (GMU) to establish a SME KB authoring laboratory at GMU's Prince William County, Virginia campus. Eight (mostly graduate) biology students participated in the TKCP evaluation, four

[1] See http://ksi.cpsc.ucalgary.ca/KAW/.

[2] More detailed tool descriptions appear in appendices. Here we include the briefest salient sketches.

working with Cycorp's KRAKEN tools, four with SRI's SHAKEN tools. All worked full-time from mid-May until mid-July, 2001. The first week of this period was devoted to classroom-style training of SMEs by teams. The next two weeks were taken up with an evaluation dry run that included shake-down of tools in the installed context and limited additional, informal training. The evaluation-proper was held during the TKCP's final four weeks. It covered about seven pages of the textbook and included 70 TQs (about 3 pages and 10 TQs having been covered in the dry run). The actual test material covered five subsections of the textbook's target material. SMEs were allowed to author this material in any order they liked, but IET would not release one subsection's TQs to a SME until s/he had completed work on TQs for earlier subsections.

Subsequent to training, SMEs had no direct contact with the teams' KEs. Instead, to deal with tool understanding issues that might arise, IET staffed the SME lab full-time with a "gatekeeper" KE who mediated contacts with the teams (including bug reports and fixes). The gatekeeper KE also provided a subjective window on SME activity. Teams were allowed to augment KBs during the evaluation in accordance with the TKCP's pump priming ground rules.

Besides these SMEs, two KEs from each team also participated (off-site from the SME lab) by addressing the same KB authoring tasks using tools of their choice. SRI KEs used the same SHAKEN tools available to the SRI-assigned SMEs. Cycorp KEs usually authored knowledge in CycL (a KE-oriented knowledge representation language) using a text editor, rather than with the SME-oriented tools in KRAKEN. Cycorp KEs did not author all target textbook knowledge during the evaluation. Instead, they relied on a base of target knowledge that Cycorp had first developed in support of its internal pump priming requirements identification, then excised before tool delivery to SMEs. (This was due to unavoidable personnel overlap between Cycorp's pump-priming and TKCP-participating KEs.) SRI KEs were given the same option but elected to author the textbook knowledge during the evaluation. All KEs authored TQ representations and developed answers independently.

SMEs and KEs participants were required to answer at least 75% of the TQs presented for each subsection. In the results below, we include for each subsection the 75% of each participant's answered questions with the highest overall scores, padding with 0s as necessary. One of these subsections ("Signals in DNA Tell RNA Polymerase Where to Start and Finish") was particularly troublesome for the Cycorp SMEs. After they had spent well over a week working on it and were all well less than halfway to reaching their answered-TQ quota, IET asked them to proceed to the next subsection to ensure that they had the chance to address most of the target material. (The SRI SMEs had completed their work and performed reasonably well on this 25-TQ subsection.) Because of this gatekeeper KE intervention, the authors have by consensus excluded this subsection from results analyses below.

Our functional performance scoring is both manual and subjective. We employ multiple scorers with expertise both in knowledge representation and in biology. We have historically achieved highly consistent results by articulating specific, value-by-value scoring guidelines for all criteria against the following, relatively coarse, generic framework: 0—no serious effort evident/completely off-base; 1—mostly unsatisfactory; 2—mostly satisfactory; 3—(for practical purposes) perfectly adequate. To arrive at an overall score for functional performance on a given TQ, we: threshold scores for the last two ancillary criteria so that they do not exceed the highest score for (one of) the earlier, primary criteria; average scores for each criterion within a category; then average the category scores.

6 EXPERIMENTAL RESULTS

6.1 Functional Performance Results

The major functional performance results are reflected in Table 1.

Team	*User type*	Representation	Answer	Explanation	Overall
Cycorp	*SME*	1.66	2.46	2.30	2.14
Cycorp	*KE*	2.54	2.58	2.56	2.56
SRI	*SME*	1.84	2.12	2.08	2.01
SRI	*KE*	2.09	2.48	2.40	2.32

Table 1: Means of teams' KEs'/SMEs' means of TQ scores

KEs' performance (the "gold standard" from RKF's perspective) was better than SMEs' with high statistical significance, but SMEs performed within 90% of the level achieved by their teams' KEs. We take the latter to reflect the relative effectiveness of teams' SME lab-fielded technology. There was no statistically significant difference across teams between the averaged scores of respective SMEs or KEs—either overall or at the criterion category level.

In a more detailed (unpublished/available upon request) treatment, we note statistically significant interactions among scores along the dimensions of individual SMEs, subsections, and question types in a categorization. All of these interactions washed out in the overall scores. We also note a "ceiling" effect, in that answer scoring with respect to several individual criteria exhibits large proportions of (highest-score) 3s. Elements likely contributing to this ceiling include our consistently accessible (i.e., low) quiz-level TQ difficulties and SMEs' consistent efforts to develop (supporting knowledge and) high-quality answers before moving on to additional TQs.

6.2 Economic / Reuse Results

Cohen et al. [2] profiled HPKB knowledge reuse as the fraction of knowledge items previously existing in a given context. We again have two main reuse contexts to explore: that of constants in axioms and that of axioms in the

explanations/proofs of answers to TQs. (To economize, we include only the latter analysis.)

Axiom reuse results appear in Table 2 Results are given for each participant (designated by monikers). Each participants' (KB's) mean overall score and mean number of axiom occurrences used to answer a TQ are included here for reference. "UA" stands for "User-authored Axioms" and "PDA" for "Pre-Defined Axioms." "Used" indicates that the noted number of axioms actually appears in the explanation to one of the participant's answered TQs. "Unused" pertains to user-authored axioms that are not so used in a TQ (e.g., because the participant used them to author a subsection's target material before receiving its TQs). "Reused" pertains to user-authored axioms used to answer more than one TQ.

Table 2: Reuse data by SME/KE

Team	*Type*	*Moniker*	Mean Overall Score	Functional Performance TQ count	Reuse TQ count	UA used	UA reused	UA unused	PDA used	UA: reused / used	UA: used / (used+unused)	Used: PDA / (PDA+UA)
A	*SME*	*Tweety*	2.17	39	31	100	34	1881	103	27.16%	10.14%	47.65%
A	*KE*	*MW*	2.85	43	35	111	13	102	150	11.71%	52.11%	42.53%
B	*SME*	*Amoeba*	2.29	30	30	538	32	1355	304	15.81%	31.75%	62.41%
B	*SME*	*Celula*	1.68	25	25	143	58	490	465	73.71%	33.65%	27.99%
B	*SME*	*Iflu*	2.26	34	34	100	53	2211	323	41.89%	11.47%	36.20%
B	*KE*	*PN*	2.58	35	35	166	132	462	313	90.72%	54.94%	53.32%
B	*KE*	*AS*	2.50	34	34	296	106	1086	340	52.27%	33.50%	57.80%
B	*SME*	*Vaccinia*	2.59	35	35	254	163	1526	383	71.84%	17.75%	47.16%

Table 2 includes only one each KE and SME entry for Cycorp because of difficulties at evaluation time with KB instrumentation and later with information extraction. These reuse results are still incomplete, as may be noted by comparing the numbers of TQs answered/scored for Functional Performance and numbers of TQs scored for reuse. A further issue of note is that the Cycorp KE, cycMW, (legitimately) authored much general knowledge directly into Cyc, as pump priming, where it is counted as pre-defined rather than user-authored.

We present (in Table 2's last columns) three varieties of reuse percentages: of user-authored axioms that appear in more than one TQ; of user-authored axioms that appear (at all) in TQs; and of appearing pre-defined out of all appearing axioms. From an economic standpoint, we comment merely that the latter reuse rate seems (uniformly) sufficiently high to justify the claim that relevant prior content has significant benefit for KB development.

We had an additional motivation (beyond economics) to examine reuse of user-authored axioms across TQs. RKF's functional performance evaluation criteria, being TQ-based, could not address the generality of knowledge across different TQs. Evaluators were interested in quantitative metrics of cross-TQ axiom reuse as a hedge against unprincipled, one-shot axiom "hacks" without lasting value.

Table 3 reports numbers of TQ occurrences for each reused user-authored axiom.

Team	*Type*	*Moniker*	TQs > 32	TQs in [17 32]	TQs in [9 16]	TQs in [5 8]	TQs in [3 4]	TQs = 2
Cycorp	*SME*	*Tweety*	0	0	0	0	2	20
Cycorp	*KE*	*cycMW*	0	0	0	0	5	8
SRI	*SME*	*Amoeba*	0	0	2	2	5	86
SRI	*SME*	*Celula*	0	0	1	69	28	59
SRI	*SME*	*Iflu*	0	0	0	7	53	51
SRI	*KE*	*sriPN*	0	0	3	186	67	57
SRI	*KE*	*sriAS*	0	1	6	97	57	81
SRI	*SME*	*Vaccinia*	0	1	1	20	99	106

Table 3: Incidences of axiom occurrence counts across TQs

Superficially, high axiom TQ-incidences occurred much more frequently for users of SRI's SHAKEN tools than for Cycorp's KRAKEN tools. (The axiom TQ-incidence patterns for pre-defined axioms are qualitatively similar.) However, these data do not appear to indicate cross-team differences in knowledge generality. Cycorp SME Tweety's axiom TQ-incidence profile is quite similar to that of Cycorp KE cycMW whose work—with highly respected representations—received the highest mean overall functional performance score. Axiom TQ-incidence profiles are also similar across SRI's KEs and SMEs. We tentatively attribute the cross-team profile differences to: compactness of (arbitrary-arity) CycL relations compared to binary relations resulting from translating SHAKEN's frames for axiom-counting purposes (suggesting a scaling factor for axioms counted in a given TQ); and conceptual coarseness, compared to pre-defined predicates in Cyc, of SHAKEN's built-in relations (leading to greater applicability across TQs). Thus, we find no overall quantitative pattern indicating deficiency of appropriate knowledge generality for either team.

6.3 Subjective Qualitative Results

Deficiencies Identification**:** The KB Quality Review Panel concluded that SMEs, working alone, performed quite well at selected KB authoring tasks, but were less effective at others. SMEs with both teams were generally adept at placing and choosing concepts from the pre-existing ontology (i.e., they created and used knowledge at correct levels of specificity) and at general process description (i.e., they implemented Event-Actor vocabulary with accuracy and ease). The Panel highlighted as shortcomings in SME KBs the following major types: incompleteness,

redundancy, and non-reusability. After describing these deficiency types below, we take up the question of their sources in the tools and in the KB authoring task.

Both teams' SMEs' KBs exhibit incompleteness, of three different kinds: content incompleteness (failure to describe a process fully, even where the textbook had); hierarchical incompleteness (failure to include natural siblings of a created concept); and interconnectedness incompleteness (failure to articulate obvious relationships between concepts).

SHAKEN SMEs' KBs exhibit significant redundancy attributable to limitations of the evaluated tools' inability to reason about authored concepts from the several distinct perspectives called for by different TQs. KRAKEN KBs also exhibit some redundancy. This usually is not of SME-authored knowledge, owing rather to re-creation by SMEs of pre-defined concepts.

Both teams' SMEs' KBs included concepts of suspect reusability. Mainly these were predicates, attributes, or concepts that combined concepts unnaturally—in a fashion that seemed difficult to reuse.

Suspected Deficiencies Sources: The Panel's and SMEs' combined attributions of the above-noted deficiencies to major tool and TKCP task sources including the following (in order of increasing challenge such sources seem likely to pose RKF tool providers): TQ- and textbook-focused SME orientation, absence or inaccessibilityof pre-defined knowledge, limited logical expressibility, and inherently difficult representation problems. We consider these in turn below.

Some KB incompleteness (especially of the content variety) are attributable to SME's attempts to tailor authoring in anticipation of unreleased TQs or in consideration of released TQs. (I.e., sometimes authoring favored TQ effectiveness over general applicability or reuse.)

That SMEs were explicitly directed to focus on authoring textbook content may explain some hierarchical and interconnectedness incompleteness.

Some of the above-noted deficiencies resulted from incompleteness in the pump-primed KBs that they received. SHAKEN did not allow SMEs to facet general collections into collections of different kinds of collection subtypes. A SME noted that this would have facilitated clearer hierarchical placement.

While KRAKEN SMEs had access to a substantial background KB and sophisticated representation language, this potential came at a price: access tended to be at times insufficient, during other times overwhelming, thereby limiting and even hampering SME productivity and expressive possibilities. Gatekeeper KE reports and SME surveys mentioned the labor-intensiveness of what turned out to serve as Cycorp SMEs' major axiom entry mode—browsing through existing axioms to discover one (with an appropriate predicate) to use as a template for editing and assertion.

Both teams' SMEs were—by design—somewhat limited in the logical forms they could use to express knowledge. SHAKEN SMEs were unable to make many assertions that deviated from the form $(\forall x\,(Ax \supset (\exists y)\,(B\ x\ y))$. KRAKEN users had access to more logical forms via a richer vocabulary of rule macro predicates, though interface issues again caused more general rule construction to be prohibitively difficult here.

A major indication frequently occurring in both team's KBs of inherently difficult representation problems is predicates lacking specificity, argument types, or supporting axiomatization. Another indication is impoverished versions of assertions whose formal representation would require complex logical expressions.

Feasibility Assessment: While it is clear that plausible near-term improvements to these tools (and their captured background knowledge) could address some of the above-noted shortcomings, it also seems (to the present authors) that KB authoring generally does include inherently difficult representation problems whose solution demands well developed logical skills and balancing different engineering principles. The ambition reflected in the present experiment to create tools that can empower a SME to full KB authoring independence—in arbitrary contexts—appears yet too grand.

While we have clear evidence that SMEs can author some high-quality knowledge in a sophisticated domain, we lack evidence that they can author high-quality predicates, analyze and refine background knowledge, develop rule paths to make sophisticated inferences work, or develop complex logical expressions required for some assertions. Also, it is not obvious how the existing tools could be refined to address such requirements.

Recommendation: We suggest that the KB development community's focus ought not be on tools that support KB authoring by "lone" SMEs (except where authoring tasks are relatively precisely defined and tools are fielded to support SMEs in a relatively mature authoring process). On the contrary, it should be on empowering SMEs to perform those KB authoring tasks they can be empowered to perform well. We believe the nascent RKF tools demonstrate a significant advance in such SME empowerment, and we recommend that in future experimental and developmental settings the relative strengths that SMEs and KEs bring to KB authoring should be exploited in a true "mixed-skills" team—a synergistic partnership.

We have some evidence that lightly trained SMEs are capable of significantly enhancing KE efforts to provide background knowledge that will be relevant to a KB authoring task. As a sequel to the TKCP evaluation, IET conducted a separate three-week evaluation intended to allow SMEs to explore teams' tools in a less structured setting. Eight (now tool-savvy) SMEs participated in an "expert knowledge" challenge problem (EKCP), pursuing KB authoring topics related to the life cycle of the Vaccinia

virus—for which teams had authored no pump priming knowledge. An IET KE who had prepared some EKCP-supporting background knowledge development (in CycL) found that a Cycorp SME (who had not effectively authored Cyc predicates working alone) was readily able to contribute an informal specification that greatly facilitated the KE's work in extending the background knowledge to support the SME's needs.

We envision such interactions occurring throughout the KB authoring process, with SMEs and KEs contributing dynamically. The KE's role is always to perform sophisticated KB authoring tasks currently beyond SMEs' reach. We believe that the SME-feasible task set should expand naturally (in a "bootstrapping" fashion) over time, as the talents of SMEs are mined and new tools are developed to meet opportunities presented by existing tools and authoring processes.

7 DISCUSSION / CONCLUSION

All styles of evaluation are useful in different contexts. Quantitative metrics are genuinely valuable for some purposes—e.g., inspiring a friendly competition among groups working in a common research initiative or demonstrating progress to an uninitiated, numbers-oriented supervisor. By far the long pole in the evaluation tent, however—from a system/process engineering, diagnostic point of view—remains subjective qualitative assessment. This is borne out by the comparative substance of our offered conclusions based on this activity and by the incorporation of insights and adoption of suggestions by technology providers working to develop the next generation of SME-empowering KB authoring tools.

We have seen that all three evaluation styles used here complement one another. The different quantitative metrics assist in each other's mutual interpretation (as, for example, when we appeal to Functional Performance in understanding Reuse), acting together as a synergistic set of reinforcements and consistency checks. We expect our effectiveness in the overall KB authoring enterprise to grow as the collective body of such techniques for understanding quality issues in KB artifacts, tools, and process continues to mature in a science of knowledge development.

8 REFERENCES

[1] Alberts, B.; Bray, D.; Johnson, A.; Lewis, J.; Raff, M.; Roberts, K.; and Walter, P. *Essential Cell Biology: An Introduction to the Molecular Biology of the Cell*, ch. 7, pp. 211–222, Garland Publishing, Inc., 1998.

[2] Cohen, P.; Chaudhri. V.; Pease, A. and Schrag, R. "Does Prior Knowledge Facilitate the Development of Knowledge-based Systems?" In *Proceedings of the Sixteenth National Conference on Artificial Intelligence*, pp. 221–2266, 1999.

[3] Cohen, P.; Schrag, R.; Jones, E.; Pease, A.; Lin, A.; Starr, B.; Gunning, D., and Burke, M. "The DARPA High-Performance Knowledge Bases Project." *AI Magazine* 19(4):25, 1998.

[4] Gruber, T. "Toward Principles for the Design of Ontologies Used for Knowledge Sharing," Technical Report, KSL-93-04, Department of Computer Science, Stanford University, 1993.

[5] Gruninger, M.; and Fox, M.S. "Methodology for the Design and Evaluation of Ontologies," in Proceedings of the IJCAI-95 Workshop on Basic Ontological Issues in Knowledge Sharing, 1995.

9 ACKNOWLEDGEMENT

This research is sponsored by the Defense Advanced Research Projects Agency (DARPA) of the United States. This document has been approved for public release, distribution unlimited.

APPENDIX A: CYCORP'S KRAKEN TOOLS

After over fifteen years of common sense knowledge base building, the Cyc project is well-equipped for Cyc to actively assist in its own extension. With its large set of common classes and instances, relationships and rules and knowledge contexts, Cyc has commenced its supervised learning process, pushing the envelope of what it knows. And, as learning occurs on the fringe of existing knowledge, leveraging and reusing this knowledge is key. This approach is realized in the KRAKEN system.[1]

The KRAKEN team metaphorically framed the task of extending a large knowledge base by viewing Cyc as akin to a child with limited proficiency in English. This (very young) "child" speaks CycL, a first-order predicate calculus-like language, as its "mother tongue" and has some knowledge of the common world. It has rudimentary notions of English, enough to verbalize most of its beliefs clearly, and can read simple English sentences with occasional help. In this view, the SME becomes a "teacher" who engages in a dialog with KRAKEN and exploits analogy, disambiguation dialog, and knowledge expectations to extend the system. Concurrently, the SME teaches the system how to express new information in English.

The KRAKEN team identified several key KB authoring tasks that KRAKEN could assist with: locating existing knowledge; adding knowledge through cut-and-paste techniques; fulfilling explicit knowledge expectations; reading simple sentences; deducing relations from examples; and assembling structured knowledge components (*e.g.*, non-trivial queries and rules) from short described scenarios. In addition, KRAKEN helps with correctness verification and strengthening of new knowledge.

For a KB of the size of Cyc, locating pieces of existing knowledge is a task in itself. At this writing, Cyc encompasses about 1.1 million assertions constructed from over 120,000 concepts and 5000 relations. Such dimensions make any "list-them-all" approach to searching impossible. However, Cyc also knows over 16,000 English verbs and nouns and over 2000 proper names and can therefore offer a natural language index into its knowledge. Once "within the vicinity" of particular concepts and relations, browsing is feasible. Additional organization is provided by Cyc's knowledge contexts ("microtheories").

Once a SME knows upon which pieces of knowledge to build, the KRAKEN system provides multiple ways for the learning process to proceed.

One principled approach is the explicit representation of knowledge expectation—*e.g.* knowing that when told of a new artist to ask for famous works by that artist. As new terms are introduced, KRAKEN will ask the SME concrete, salient questions. This approach is especially interesting, as the KRAKEN team is adding support for a SME to teach the system such knowledge expectations as well.

One major goal of the KRAKEN team has been to support KB authoring using simple English sentences. KRAKEN parses the sentences into an underspecified representation, which is then reformulated, based on the analysis of applicable argument constraints, into CycL. During reformulation, KRAKEN attempts to solidify the quantification, an aspect vital to knowledge engineering and highly ambiguous in natural languages. (Compare the class-level statement, "A dog is a mammal," to the instance level statement, "A dog is in the yard.") Like anyone learning English, KRAKEN asks for help when it gets stuck.

KRAKEN ensures—within bounds of reason—that the new information is semantically valid and neither in contradiction nor redundant with existing information. Even more important, KRAKEN attempts to fine-tune the strength of statements by suggesting ways to change their specificity or generality. Since stating knowledge at the correct level of generality requires mastering the available alternatives, KRAKEN guides the SME to subsumed or covering statements. This approach also exploits the human ability of recognition, instead of relying on recall.

No predicate set is ever complete, and KRAKEN provides the ability for the SME to define new relationships. The acquisition paradigm is structured around use cases: the SME provides KRAKEN with examples of how the predicate will be employed. This not only allows KRAKEN to compute the new relationship's argument constraints automatically but also jump-starts the population of the relationship and provides KRAKEN with believed suitable exemplars for communicating these relationships to other users.

For the assembly of more complex knowledge constructs, such as non-trivial queries and implications, the KRAKEN team has chosen an almost story-like approach: the SME lays out a scenario for KRAKEN, consisting of the involved terms and the relationships between these. Once the scenario has been "narrated" in this fashion, KRAKEN assembles the relationships and terms into a query or an implication.

[1] In order to achieve this goal, Cycorp teamed with Hans Chalupsky at the University of Southern Clifornia's Information Sciences Institute, Ken Forbus' Qualitative Research Group at Northwestern University, and the Artificial Intelligence Applications Institute at the University of Edinburgh to construct the KRAKEN system around Cyc.

APPENDIX B: SRI'S SHAKEN TOOLS

The claim of the SHAKEN effort is that SMEs, unassisted by AI technologists, can assemble models of mechanisms and processes from components. These models are both declarative and executable, so questions about the mechanisms and processes can be answered by conventional inference methods (for example, theorem proving and taxonomic inference) and by various task-specific methods (for example, simulation, analogical reasoning, and problem-solving methods). A related claim is that relatively few components, perhaps a few thousand, are sufficient for SMEs to assemble models of virtually any mechanism or process. We claim that these components are independent of domain, and that assembly from components instantiated to a domain is a natural way for SMEs to create KB content.

The research in this project exploits and extends previous work in KBs, process description languages, qualitative physics, systems dynamics, and simulation. One scientific innovation is the idea of declarative and executable models (DEMs) assembled from components. The declarative aspect of DEMs supports conventional inference, whereas the executable aspect supports reasoning by simulation. For example, the declarative part of a model of aerosols is sufficient to answer questions like, "Will a 5-micron filter afford protection against this aerosol?" while the executable part is necessary to model the dispersal pattern of the aerosol.

The development of libraries of components made available to SMEs via restricted natural language based, graphical, or templatized interfaces is the principal means by which logic-oriented knowledge representation formalisms become accessible to ordinary users. Every modeling technology shows this progression: spreadsheets, finite-element packages, statistical packages, chemical synthesis software, Macsyma and Mathematica, architectural and CAD packages, graphics and HCI systems, *etc.* are accessible to ordinary users because they offer libraries of components. As a practical matter, then, it makes sense to provide SMEs with libraries of modeling components. As a scientific matter, we believe we can develop components that represent how humans think about mechanisms and processes.

The SHAKEN system has the following major functional components: a knowledge base, an interface for entering knowledge and asking questions, and a knowledge server.

The KB, also called the component library, contains a collection of components representing (1) general knowledge about common physical objects and events, states of existence, and core theories, including time, space, and causality, and (2) more specialized knowledge about microbiology and biological warfare agents. By a "component," we mean a coherent set of axioms that describe some abstract phenomenon (*e.g.*, the concept "invade") and that are packaged into a single representational unit.

The SHAKEN KB evaluated here contained roughly 250 components representing domain-independent events. These components would make copious use of core theories of time, space, and partonomy [7].

A graphical interface for knowledge entry enables a SME to assemble KB components. By "assembly," we mean the connection of components from the component library. The system evaluated here supports four basic operations: "connect," "specialize," "unify," and "add" [7]. Axioms are derived from the graphical representation, and the SME does not have to be trained in formal logic. The graphical representation is created by a combination of manual and automatic means.

The question-asking interface plays a central role in knowledge entry. A SME must be able to understand what is already encoded in the system, to locate components for assembly, and to ask arbitrary questions. SHAKEN returns answers in an easily understood format, and a SME is able to control the level of detail in an answer. SHAKEN as evaluated here supported parameterized questions—derived from a viewpoint grammar [6]—and similarity search. Presentation of answers to a SME is controlled using explanation design plans.

The knowledge server provides facilities for efficient storage and access, supports inference for answering questions and for assembly of components, and includes both general-purpose inference and special-purpose inference. For SHAKEN as evaluated here, reasoning support was provided by the Knowledge Machine (KM) representation system.

REFERENCES FOR APPENDIX B

[6] Acker, L. and Porter, B. "Extracting Viewpoints from Knowledge Bases," in Proceedings of the Twelfth National Conference on Artificial Intelligence Annual Conference (AAAI-94), pp 547–552, 1994.

[7] Barker, K., Porter, B., and Clark, P. "A Library of Generic Components for Composing Knowledge Bases," in proceedings of the International Conference of Knowledge Capture, 2002.